NISTIR 89-4199

GRAPHICS APPLICATION PROGRAMMER'S INTERFACE STANDARDS AND CALS

Sharon J. Kemmerer
Mark W. Skall

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
National Computer Systems Laboratory
Gaithersburg, MD 20899

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Raymond G. Kammer, Acting Director

NIST

NISTIR 89-4199

GRAPHICS APPLICATION PROGRAMMER'S INTERFACE STANDARDS AND CALS

Sharon J. Kemmerer
Mark W. Skall

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
National Computer Systems Laboratory
Gaithersburg, MD 20899

October 1989

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Raymond G. Kammer, Acting Director

ABSTRACT

The principal purpose of a graphics Application Programmer's Interface (API) standard is to provide portability for an application program across a wide range of computers, operating systems, programming languages, and interactive graphics devices. The graphics API's are represented by four major standards' projects: Graphical Kernel System (GKS), GKS-3D, Programmer's Hierarchical Interactive Graphics System (PHIGS), and Programmer's Imaging Kernel (PIK).

This deliverable provides background information on past graphics API recommendations; introduces the concepts of an API; focuses on the two standards of interest to CALS: PHIGS and PIK; describes CALS applications for these standards; and recommends to the CALS Office the utility of graphics API standards.

TABLE OF CONTENTS

GRAPHICS APPLICATION PROGRAMMER'S INTERFACE STANDARDS AND CALS

I. PURPOSE.

This is a Computer-aided Acquisition and Logistic Suppport (CALS) Program deliverable in response to tasks 4.1.1 and 4.2.1. (NIST Statement of Work dated March 3, 1989). It contains a discussion on graphics Application Programmer's Interfaces (API's) in general, with specific focus on two current API standard initiatives: Programmer's Hierarchical Interactive Graphics System (PHIGS) and Programmer's Imaging Kernel (PIK).

II. BACKGROUND.

In the NIST CALS report on graphics, FY 1986 [CALS86], functional requirements for API standards were suggested specific to CALS applications. To summarize the recommendations for API use from that report:

- Engineering Design and Drafting. "The application running on the CAD workstation used to create original drawings and modify existing drawings could be built upon GKS or PHIGS--themselves, perhaps layered, on top of CGI--to provide device-independence and code portability."

- Procurement Support. "...where previewing of drawings is required, applications may want to build upon the API graphics standard to get the usual benefits of program portability, maintenance cost savings, and device-independence."

- Automated Technical Manual Systems. "The role for an API standard--either GKS or PHIGS--in this scheme is during the graphics editing/graphics enhancement stage. Applications written to GKS or PHIGS will be able to easily read in CGM metafiles, modify their contents, and write them out again."

III. DISCUSSION.

A. API Introduction.

API standards are tool boxes of graphics functions used by application programmers who develop graphics programs. Each API standard is then "bound" to all programming languages which may be used in conjunction with that standard. Graphic API's are represented by four major standards' projects: Graphical Kernel System (GKS), GKS-3D, PHIGS, and PIK. These API standards are typically implemented as a collection of external procedures or subroutines that a programmer can link with his

application code to obtain graphical input and cause pictures to be displayed on graphical output devices.

The API standards are not directly suitable for picture exchange; however, each standard has an associated storage mechanism (an archive file or graphical metafile,) that can be used to exchange graphical information between systems using the same standard. In addition, the Computer Graphics Metafile (CGM) is a general-purpose picture transfer standard that can be created by any application and can be used in conjunction with any of the graphics standards.

The principal purpose of an API standard is to provide portability for an application program across a wide range of computers, operating systems, programming languages, and interactive graphics devices. Consequently, programs written to an API standard at one facility can be exchanged with another facility and used with only minor modifications needed to tailor the software to the implementation differences allowed by the standard.

Furthermore, as CPU's and peripherals are upgraded and replaced, software written to an API standard will survive and need not be rewritten. Indeed, the software performance should improve, assuming that the new hardware is more capable than the old hardware, and new graphics hardware will be developed taking the graphics standards into consideration [AB88].

Some other benefits from using API standards include: reducing software development and lifecycle costs, serving graphics equipment manufacturers as a guideline in providing useful combinations of graphics capabilities in a device, and reducing programmer costs and time since many of the functions currently performed by the application program will now be performed by the API standard [ANS87].

B. <u>PHIGS</u>.

PHIGS specifies an API to a rich, device-independent graphics environment. PHIGS is designed to support such important applications as CAD/CAE/CAM, command and control, molecular modelling, simulation, and process control. PHIGS emphasizes the support of applications needing a highly dynamic, highly interactive operator interface and expects rapid screen update of the complex images to be performed by the display system.

PHIGS provides all the viewing capabilities of GKS-3D in a compatible manner, but, in addition, PHIGS supports the creation, modification, and viewing of a geometric model, which is maintained by the PHIGS implementation. Stored in an area called the Centralized Structure Store, PHIGS elements are structured into hierarchies, with structures calling other

structures and with offspring structures inheriting attributes from parent structures. Once created, or while being created, PHIGS structures can be marked for display on one or more workstations. A powerful feature of the system is the provision to allow scanning and selective editing of the contents of the stored structure, which in turn allows the results of interactive sequences to be displayed without completely redefining all the displayed structures.

C. PIK.

1. Introduction.

PIK is a new standard initiative within Accredited Standards Committee X3H3 (specifically subcommittee X3H3.8). The Committee was formally established in the fall of 1988, and is hoping to "fast track" the standard's development process. An ambitious goal of developing a draft ANS is set for December 1991. Although specific levels of detail for the scope of PIK have not yet been determined by the Committee, general information and PIK's applicability to CALS are discussed in this report. The information provided here draws heavily from the current PIK strawman [SM3-89].

2. Scope and Purpose.

PIK is an API for electronic imaging. It defines the data objects and a set of functions which can be applied to those objects for use in imaging applications. The PIK standard supplies the basic building blocks upon which applications requiring imaging functionality can be built within conventional, distributed, and imaging-oriented computing environments.

The intent of the PIK standard is to provide application developers with a rich set of imaging services without affecting the architecture used to implement or provide those services. PIK is intended to support operations on various classes of images from a wide variety of imaging applications, including those which simply display images, to those which require highly interactive, human-directed image processing. The PIK standard provides the means for the sharing and use of specialized display architectures and image processing hardware.

3. Functional Model.

The purpose of a functional model is to provide the reader with a mental framework or roadmap against which the proposed standard can be interpreted. Not surprisingly, such a model has multiple dimensions which provide

different perspectives. The Task Group is still evaluating which proposed reference model provides the best description and intent of PIK. For the purposes of this document, the functional model shown in Figure 1 has been chosen to emphasize PIK's relationship to other standards already developed or under development.

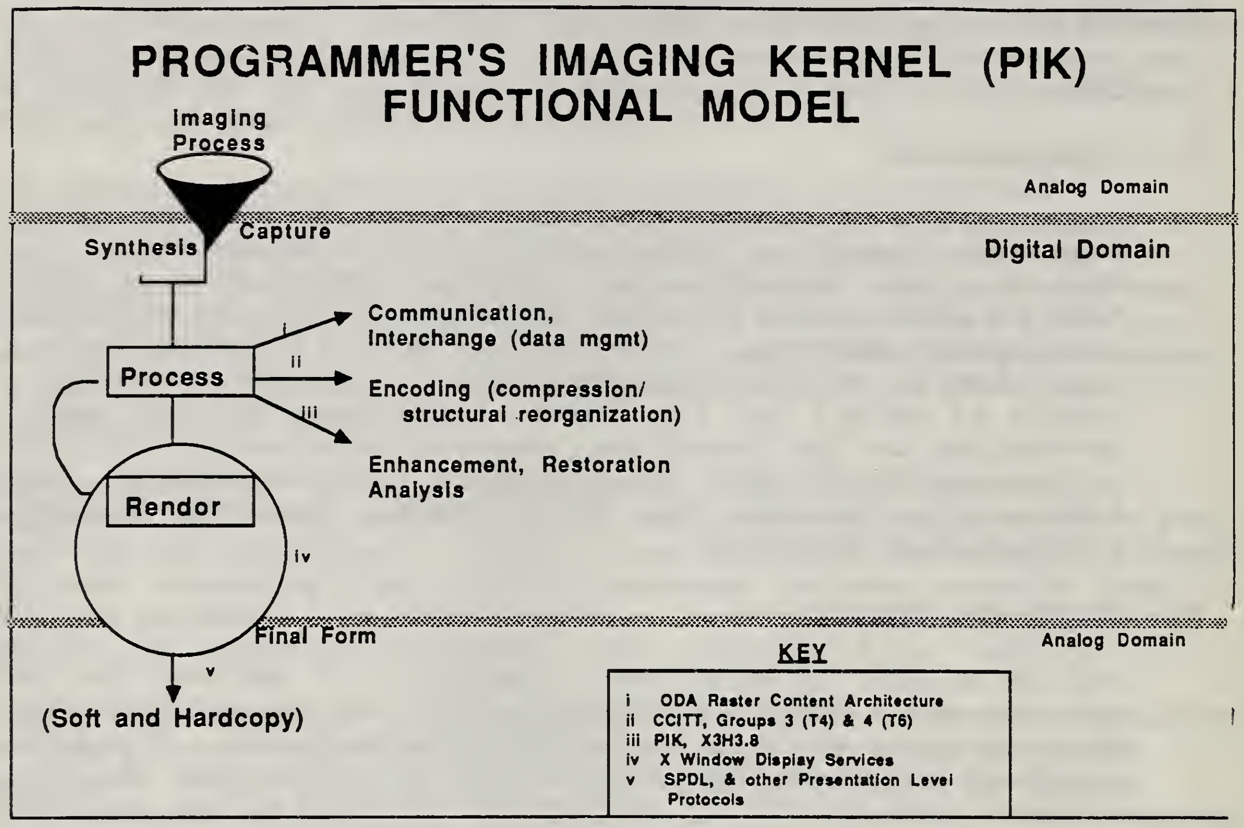

Figure (1)

4. Operational Concepts.

Graphics and imaging operations or processes were conducted for inherently different purposes. Graphics, as currently defined in the standards, is used for scene construction, modification, and viewing (presentation). Imaging, as defined by current practice, is used for modification (refinement processing), viewing (presentation), and for extractive or analysis ends. Today overlap exists between graphics and imaging during manipulation, and at or near the presentation (viewing) stage, especially in the concept of a screen space, or a viewable region.

5. Graphics and Imaging Comparison.

A distinction between imaging and graphics can be made at the beginning: imaging starts with an image taken from the

real world (a photograph or hardcopy) and pulled into the system; a graphical representation is a derived object from scientific calculation. Images are more input/output bound, whereas graphics development is very numerical in process and more CPU time-oriented. Once both images and graphical objects are created or pulled into the system, the distinctions between the two blur. Normal operations of manipulation (e.g., scaling and rotation) and analysis were once considered solely for graphics. Today, these same types of functions occur on images as well.

Often, when dealing with image processing applications, it is desirable to combine bits and pieces of imagery and graphics to form a composite final display. One may have a need for retrieving and overlapping an image with a graphical object. The graphical object is still derived mathematically, the image is derived from the real world.

IV. GENERAL APPLICATION TO CALS.

A. PHIGS.

Typical CAD/CAM programs, written using PHIGS to obtain device independence, will build internal geometric models and data structures by loading data from an external product definition data path (e.g., IGES or STEP/PDES). During the processing, CGM picture files may be produced for later use in such applications as computer-assisted publishing and picture previewing [AB88]. IGES files may be produced as output for engineering drawing applications. Outputting CGM and IGES files allows picture data and engineering data to be captured and sent to other processors, perhaps at other locations for further manipulation.

There is a current effort underway to make PHIGS and IGES/PDES/STEP much more consistent and compatible. In fact, PHIGS is being promoted as important foundation material for use in STEP/PDES, particularly for Presentation attributes and presentation functionality. The following excerpt was taken from the international ISO Presentation Committee minutes, IGES/PDES/STEP Organization meeting, San Antonio, Texas, April 10-14, 1989:

> "A thorough discussion of the issues related to graphics led to a clarification of underlying STEP specific requirements and possible solutions.
>
> With regard to the use of neutral international graphics standards as a tool for Presentation the following policy was adopted... The concepts of graphical standards will be used by Presentation whenever possible." [SAN89]

International ISO discussions of particular interest to CALS include the use of PHIGS functions for such presentation entities as structure, geometry, symbols, and text and fonts [GSC89]. Additionally, PHIGS and CGM are based on the same graphics model developed by ASC X3H3. They use essentially the same primitives and attributes. Thus CAD/CAM programs based on PHIGS can produce CGM and IGES files more efficiently and more economically than programs using another graphics package or native graphics languages.

Although data portability within CALS is a worthy goal, a far more substantial investment in application development needs to be preserved if the applications software is to be delivered to and used by DOD. For example, concept and preliminary design programs, even if developed under contract, would be used by DOD. Mandating that CAD/CAM applications, such as these, use PHIGS to do their graphics manipulations will allow these programs to run on a variety of computers and graphics devices both at DOD and contractor sites. Furthermore, when more sophisticated graphics devices replace the current ones, the PHIGS-based CAD/CAM programs will utilize the new devices with no modification to the application software.

B. <u>PIK</u>.

Although there are many other industries with applications for image processing, the following fields of interest have been selected for discussion. Some of these areas are more closely related to the current needs of CALS environments identified today. Other areas are provided as educational information for future reference. As CALS continues to evolve and horizons for the phases are defined and refined, some of the uses for PIK identified here may begin to play a key role.

The presentation of this information will define the "market," that particular market's data requirements, and the associated functional requirements. The lists and requirements are not considered exhaustive, but are presented as concepts for thought.

<u>Remote Sensing</u>. This market area is one of the oldest and most mature applications of image processing. Remote sensing is the enhancement and analysis of images obtained from a distance. The camera, normally an extremely high resolution photographic video or sampling device, produces a picture which is then digitized. Typical applications are in seismic work, mapping, and military applications. <u>Data requirements</u>. Very large quantities of data are generated, and many users demand a minimum of 2Kx2K viewable pixels. <u>Functional requirements</u>. Filtering, image composition, feature extraction, segmentation.

Industrial Vision. This market segment is a relatively new application area of image processing technology to manufacturing problems. Industrial vision products are systems which can capture visual data and make decisions based on the informational content of that data for tasks such as inspection/testing and processor control. Inspection tasks include inspection of products for fault testing during assembly and quality assurance. Other industrial vision systems employ their reasoning abilities to control other devices such as robots. Data requirements. Typical applications require 8 bits or less of monochrome image data, and 480 lines or less of image spatial resolution. Functional requirements. Generally performed in real time, some types of imaging functions required by inspection and control include: image acquisition, image alignment functions (rotation, transformation, warping), statistical analysis, line or area profiling.

Electronic Publishing. Image processing systems and techniques are used in the electronic publishing market for the creation of inputs for printing presses, optical character recognition (OCR) front-ends to technical publications systems, and for merging image data with graphics and text. From an image processing technology point of view, the demands of this market are not heavy. Rather, the push in the market is for scanning devices that can scan an image at or near photographic resolution, in order to displace much of the manual film-based work that is dominant in the publishing area today. Many major publications are composed from scanned inputs, keyed test, and graphic objects, which are matted, color tweaked and airbrushed, color separated, screened, and printed in a customized computer solution. Data requirements. Very high spatial resolutions are required in order to support the high resolution output devices. Typically, one bit per pixel is sufficient for capturing textual information; gray-scale and color data are used for graphics and images. Functional requirements. Complex algorithms for synthesizing information from image data are typically not required. Instead, the types of imaging functions required by the electronic publishing market include: image acquisition, editing and composition; point operations; page rotations (portrait/landscape); manipulation of raster text; OCR; and halftoning.

Graphic Arts. This market has grown with the application of computers to assist or replace manual paste-up graphic arts activities. Further, this shift has started with largely computer graphics capabilities and is expanding into mixed graphics, text and imaging. These systems must provide extremely friendly and natural user interfaces to designers and artists, and advanced graphics capabilities to allow for the creation of images with a very high number of real colors. Data requirements. Serious graphic arts applications require

several to many colors, and moderate to high spatial resolution. Convenient scanned input and appropriate level output are essential. Functional requirements. Types of imaging functions required by the graphics arts market include: image acquisition; interactive image editing, brushing and painting; annotation with quality raster fonts and symbols; and color transformations.

Automated Manufacturing. With the more routine use of computerized design and robotic manufacturing, image synthesis is becoming a more practical approach to product quality control. Corporations are using image processing to expose flaws, burrs, and other irregularities in products as they are coming off the assembly line. Data requirements. 2D or 3D high resolution gray scale or color images. Functional requirements. Such a market's requirements would include: color transformations, line or area profiling, image composition and feature extraction. [OP89]

Mission Planning. Primarily associated with DoD and Government, the requirement is to plan and/or rehearse military missions using integrated computer software and hardware to allow interactive viewing of satellite photos, map features, threat placement, etc. The generation of perspective views of the terrain and threats as they would appear from a low altitude is a popular requirement. Data requirements. Moderate 2D or 3D spatial resolution with many colors are normally required in this market. Functional requirements. Proper registration of independent image and graphic components on the screen is an important function. Perspective view generation can be performed as a shaded polygonal rendering (graphics), as an image warp, or as a 3D image ray trace problem. Other types of imaging functions required in the mission planning scenario include: image and graphic object editing, filtering operations, and measurements.

V. PHASE I CALS APPLICATIONS AND API STANDARDS.

Figure (2) below, graphically depicts the relationships between currently adopted CALS standards under Phase I planning; API standards developed or under development by ANSI; and specific applications against which both apply and interact. Three different types of CALS applications are affected: engineering drawings (lifecycle support of weapons systems), raster images (repository data, technical illustrations), and technical publications (technical manuals, training manuals, and operation guides).

Within CALS, the graphics within each of these application areas use different standards to store the data representation engineering drawings are represented using IGES; raster images are represented as pixels and compressed using CCITT, Group 4;

and drawings within technical publications are represented using CGM.

During the CALS life cycle, modifications need to be made to the drawings in each of the three application areas; thus, API standards are needed to preserve the software programs performing the modifications. As Figure (2) illustrates, modifications to engineering drawings are performed using sophisticated CAD/CAM software which should be based on PHIGS; modifications such as pixel manipulation on a single image or blending of two raster images should use PIK; and modifications to technical publications illustrations should use GKS because these illustrations are 2-D in nature and modification programs would be relatively unsophisticated graphics programs, not requiring the overhead of PHIGS. SGML tags and identifies the parcels of text, while ODA is used to create a single formatted text/graphics data stream for exchange.

API STANDARDS AND CALS

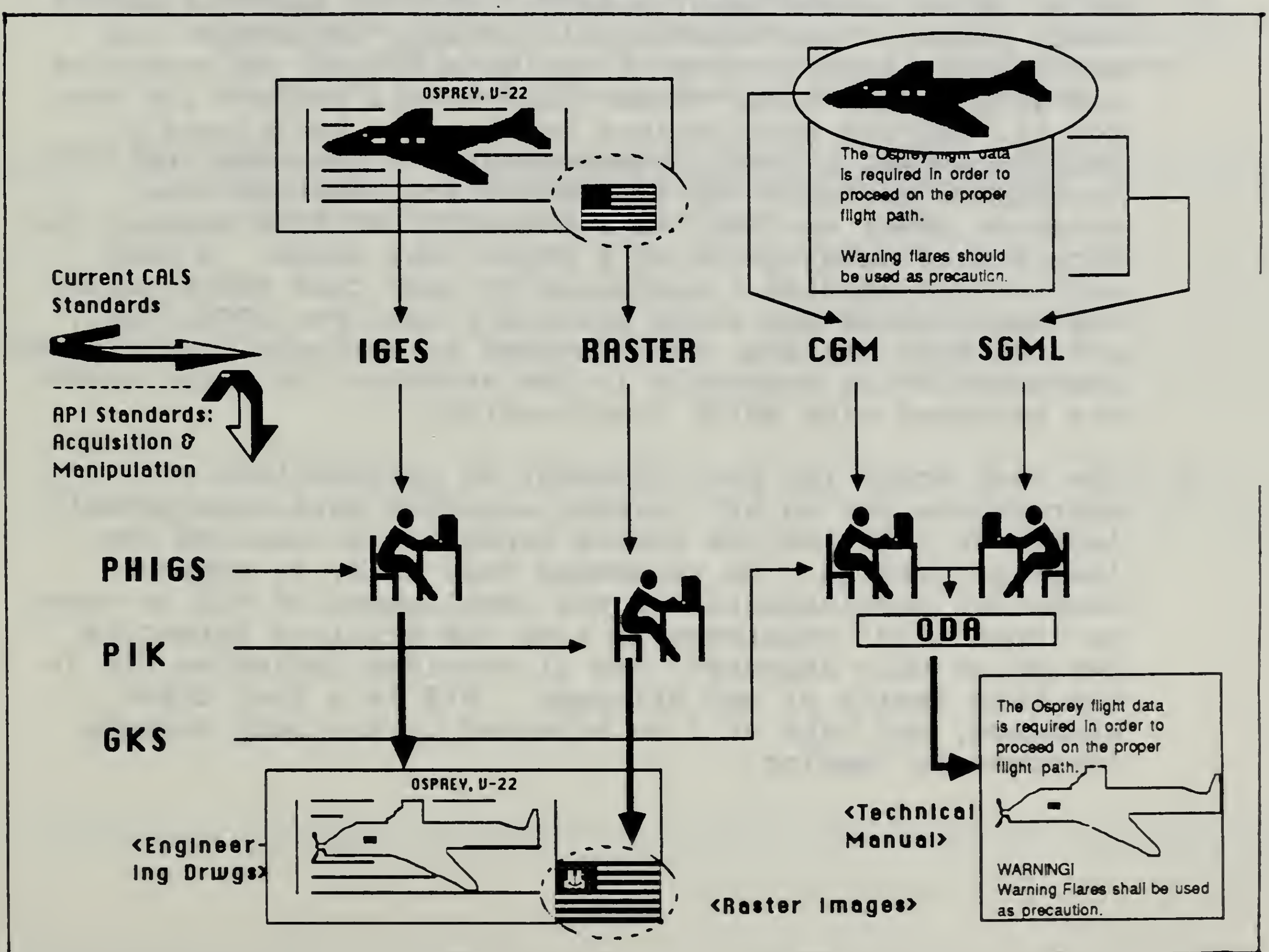

Figure (2)

API standards to exchange programs between dissimilar systems will allow interaction between repositories across all participating DoD agencies, and provide previewing capabilities of contractual deliverables. These standards can also play a critical role in precontractural stages of product and weapon system development for passing conceptual design and concept exploration (study contracts) application software between DOD activities. This stage of life cycle development is key to the success of any weapon system; cost savings and more efficient production during this stage can provide better quality information for the eventual contractor, as well as a good marketing strategy during Congressional negotiations and "sale" of a program. It establishes the foundation for standardizing access to the current data and all cumulative information that follows.

VI. RECOMMENDATION.

1. PHIGS is the only viable API standard for CAD/CAM programs. Using PHIGS in the development of CAD/CAM applications would preserve the substantial software investment and allow these applications to run on a variety of computers and graphics devices. Thus, CALS should mandate its use for all CAD/CAM applications developed. Additionally, PHIGS has been actively recognized as a contributing foundation standard for STEP/PDES, and uses the same graphics model as CGM. We recommend that CALS support the accelerated development of a PHIGS test suite. A test suite would provide a mechanism to test CALS PHIGS-based implementations and would provide a tool for PDES, Inc., and the NIST National PDES Testbed to evaluate Presentation characteristics currently in the standard, or those which are replaced with PHIGS functionality.

2. The vast array (no pun intended) of applications appropriate for an API imaging standard have considerable impact on CALS and the future capabilities desired for image processing. We recommend that CALS, 1) support technical participation in the development of PIK in order to input CALS' requirements into the critical formative period of this standard; and 2) consider including PIK in the CALS family of DoD Milspecs. PIK is a fast track standard, and CALS will be affected by the many markets involved in imaging.

VII. REFERENCES.

[AB88] Arnold, D.B. and Bono, P.R., CGM and CGI, Springer-Verlag, Berlin, Germany, 1988.

[ANS87] American National Standard X3.144-198x (Part 1), Programmer's Hierarchical Interactive Graphics System, October, 1987.

[CALS86] NBSIR 87-3566, Final NBS Report for CALS, FY86, "CALS Final Report, Graphics Standards," May 1987.

[GSC89] Carson, George, "Position Paper on General Concepts for STEP Presentation Entities," 4 April, 1989.

[OP89] Otto, Paul, "From a Captured Image to Useful CAD graphics," Advanced Imaging, PTN Publishing Co., Woodbury, NY, April 1989.

[SAN89] Nowacki, Horst, April 17, 1989 memorandum, "Report on Presentation Group Results Achieved in San Antonio at the ISO STEP Meeting," Fachbereich 12 Verkehrswesen, Institut fur Schiffsund Meerestechnik, Germany.

[SM3-89] ANSI X3H3.8, Programmer's Imaging Kernel (PIK), Strawman 3, Imaging Applications Programmer's Interface Task Group, January 20, 1989.

NIST-114A (REV. 3-89)	U.S. DEPARTMENT OF COMMERCE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY **BIBLIOGRAPHIC DATA SHEET**	1. PUBLICATION OR REPORT NUMBER NISTIR 89-4199 2. PERFORMING ORGANIZATION REPORT NUMBER 3. PUBLICATION DATE October 1989

4. TITLE AND SUBTITLE

Graphics Application Programmer's Interface Standards and CALS

5. AUTHOR(S)

Mark Skall and Sharon Kemmerer

6. PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)	7. CONTRACT/GRANT NUMBER
U.S. DEPARTMENT OF COMMERCE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY GAITHERSBURG, MD 20899	8. TYPE OF REPORT AND PERIOD COVERED

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)

10. SUPPLEMENTARY NOTES

[ ] DOCUMENT DESCRIBES A COMPUTER PROGRAM; SF-185, FIPS SOFTWARE SUMMARY, IS ATTACHED.

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

The principal purpose of a graphics Application Programmer's Interface (API) standard is to provide portability for an application program across a wide range of computers, operating systems, programming languages, and interactive graphics devices. The graphics API's are represented by four major standards' projects: Graphical Kernel System (GKS), GKS-3D, Programmer's Hierarchical Interactive Graphics System (PHIGS), and Programmer's Imaging Kernel (PIK).

This deliverable provides background information on past graphics API recommendations; introduces the concepts of an API; focuses on the two standards of interest to CALS: PHIGS and PIK; describes CALS applications for these standards; and recommends to the CALS Office the utility of graphics API standards.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

API; CALS; GKS; graphics standard; PHIGS; PIK

13. AVAILABILITY	14. NUMBER OF PRINTED PAGES
[X] UNLIMITED	16
[ ] FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).	15. PRICE
[ ] ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.	A02
[X] ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.	

ELECTRONIC FORM